

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра Информатики

Дисциплина: Операционные системы и сети

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту на тему

«Анализ и устранение уязвимостей в локальной сети»

БГУИР КП I – 40 04 01

Выполнил

студент гр. 053504

Соколовский А.А.

Проверил:

Давыдчик А.В.

Минск 2023

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
Кафедра информатики

«УТВЕРЖДАЮ»

Заведующая кафедрой

_____ Н.А. Волорова

« » _____ 2023 г.

З А Д А Н И Е

к курсовому проекту по дисциплине «Операционные системы и сети»

Группа 053504

Студенту _____ Соколовскому Аркадию Анатольевичу

(указать полностью фамилию, имя, отчество)

1. Тема курсового проекта: _____ Анализ и устранение уязвимостей в локальной сети
(указать название)

2. Сроки сдачи студентом законченного проекта: 15.05.2023 г.

3. Исходные данные к проекту:

3.1. Описание к выполнению: Разработка программного обеспечения, алгоритма с блок-схемой и написание кода программы с учетом выбранной архитектуры вычислительной системы и системного программного обеспечения.

3.2. Язык и среда программирования – C/C++/Python/C# (выбор с учетом задания может быть скорректирован). Среда программирования выбрать исходя из языка программирования.

3.3. Пояснительную записку и графический материал выполнять по СТП БГУИР 01-2017.

3.4. Другие требования уточняются студентом в процессе работы.

4. Содержание расчётно-пояснительной записки (перечень подлежащих разработке вопросов):

Титульный лист. Заполненный бланк задания с приложением. Содержание.

Введение (Актуальность темы курсовой работы; цель и перечень задач, которые планируется решить; детальная постановка задачи)

Названия пп.4.1-4.4 не являются строго утверждёнными для публикации в пояснительной записке и могут быть переименованы студентом с сохранением общего смысла.

4.1. Операционная система. (Структура и архитектура вычислительной системы. История, версии и достоинства. Обоснование выбора вычислительной системы. Анализ выбранной вычислительной системы для написания программы)

4.2 Платформа программного обеспечения. (Структура и архитектура платформы. История, версии и достоинства. Обоснование выбора платформы. Анализ операционной системы (или другого программного обеспечения) для написания программы)

4.3. Теоретическое обоснование разработки программного продукта (Обоснование необходимости разработки. Технологии программирования, используемые для решения поставленных задач. Связь архитектуры вычислительной системы с разрабатываемым программным обеспечением)

4.4. Проектирование функциональных возможностей программы (Обоснования и описание функций программного обеспечения (программы, приложения) с учетом выбранной темы курсового проекта)

4.5. Архитектура разрабатываемой программы (Общая структура программы. Описание функциональной схемы программы. Описание блок-схемы алгоритма программы.)

Заключение (Выводы по курсовой работе)

Список литературных источников (Перечень литературы и интернет-источников, которые были реально использованы при выполнении курсовой работы)

Приложения (Ведомость документов, листинг программного кода и др.).

5. Перечень графического материала:

(Форматы бумаги не являются строго утверждёнными и могут быть изменены студентом по мере необходимости)

5.1. Функциональная схема алгоритма, реализующего программное средство (листов 1, формата А3).

5.2. Блок схема алгоритма, реализующего программное средство (листов 1, формат А3)

5.3. Графический интерфейс пользователя (плакат 1, формат А3)

6. Консультант по проекту: преподаватель Давыдчик Антон Витальевич.

7. Дата выдачи задания: 16 сентября 2022 г.

8. Календарный график работы над проектом на весь период проектирования:

№ п/п	Наименование этапов курсового проекта	Срок выполнения этапов проекта	Примечание
1.	1-я опрoцентoвка (пп. 4.1, 4.2)	01.03.2023	30%
2.	2-я опрoцентoвка (пп. 4.3, 4.4, 4.5)	01.04. 2023	60%
3.	3-я опрoцентoвка (зaключение, приложения, графический материал с программным продуктом)	01.05. 2023	80%
4.	Сдача курсового проекта на проверку	14.05. 2023	100%
5.	Защита курсового проекта	15.05. 2023	Согласно графику

Руководитель

А.В. Давыдчик

Задание принял к исполнению 01.02.2023

_____ (Соколовский А.А.)
(подпись студента) (расшифровка подписи)

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	3
ВВЕДЕНИЕ	3
1 ОПЕРАЦИОННАЯ СИСТЕМА	3
2 ПЛАТФОРМА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	4
3 ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ РАЗРАБОТКИ ПРОГРАММНОГО ПРОДУКТА	4
4 ПРОЕКТИРОВАНИЕ ФУНКЦИОНАЛЬНЫХ ВОЗМОЖНОСТЕЙ ПРОГРАММЫ	4
5 АРХИТЕКТУРА РАЗРАБАТЫВАЕМОЙ ПРОГРАММЫ	4
ЗАКЛЮЧЕНИЕ	4
СПИСОК ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ	4
ПРИЛОЖЕНИЯ	4

ВВЕДЕНИЕ

В современном мире большинство компьютерных систем подключены к сети. Данный факт, повышает угрозу их поражения вредоносным ПО. Для подобных случаев существуют сетевые анализаторы. Данные приложения собирают информацию о сети, к которой подключено устройство и на основе полученных данных делают какие-либо выводы, предлагая пользователю выбор операций, которые можно выполнить. Данные операции помогут сделать сеть безопаснее.

Цель курсовой работы - ознакомиться с вариантами уязвимостей локальной сети и научиться их устранять.

Данная тема актуальна и в наши дни, каждый ПК может иметь нежелательное подключение.

Задачи: Данный код выполняется из командной строки и анализирует сеть, предлагая варианты как ее обезопасить.

1 ОПЕРАЦИОННАЯ СИСТЕМА

Linux — это семейство операционных систем (ОС), работающих на основе одноименного ядра. Нет одной операционной системы Linux, как, например, Windows или MacOS. Есть множество дистрибутивов (набор файлов, необходимых для установки ПО), выполняющих конкретные задачи.

Линус Торвальдс — первый разработчик и создатель Linux. Именно в честь него и была названа ОС. В 1991 году Линус начал работу над собственной ОС семейства Unix. Через три года появилась первая версия, доступная для скачивания. Но тогда она имела очень низкий спрос — ей пользовались буквально несколько человек.

Только через 10 лет ОС Linux получила широкое распространение. Сообщество программистов подхватило идею свободного ПО, специалисты стали помогать развивать проект.

Логотип и талисман Linux — пингвин Тух с желтыми лапами и клювом. В 1996 году разработчики ядра Linux решили выбрать талисман. Торвальдс обмолвился о том, что ему очень нравятся пингвины. Поэтому на логотипе ОС изображен пингвин.

В отличие от Microsoft Windows, Mac и коммерческих UNIX-подобных систем, GNU/Linux не имеет географического центра разработки. Нет и организации, которая владела бы этой системой; нет даже единого координационного центра. Программы для Linux — результат работы тысяч проектов. Некоторые из этих проектов централизованы, некоторые сосредоточены в фирмах. Многие проекты объединяют хакеров со всего света, которые знакомы только по переписке. Создать свой проект или присоединиться к уже существующему может любой и, в случае успеха, результаты работы станут известны миллионам пользователей. Пользователи принимают участие в тестировании свободных программ, общаются с разработчиками напрямую, что позволяет быстро находить и исправлять ошибки и реализовывать новые возможности. Именно такая гибкая и динамичная система разработки, невозможная для проектов с закрытым кодом, определяет исключительную экономическую эффективность GNU/Linux. Низкая стоимость свободных разработок, отлаженные механизмы тестирования и распространения, привлечение людей из разных стран, обладающих разным

видением проблем, защита кода лицензией — всё это стало причиной успеха свободных программ. Конечно, такая высокая эффективность разработки не могла не заинтересовать крупные фирмы, которые стали открывать свои проекты. Так появились Mozilla (Netscape,), OpenOffice.org (Sun), свободный клон Interbase (Borland) — Firebird, SAP DB (SAP). IBM способствовала переносу GNU/Linux на свои мейнфреймы. С другой стороны, открытый код значительно снижает себестоимость разработки закрытых систем для GNU/Linux и позволяет снизить цену решения для пользователя. Вот почему GNU/Linux стала платформой, часто рекомендуемой для таких продуктов, как Oracle, DB2, Informix, SyBase, SAP R3, Domino.

Дистрибутивы GNU/Linux

Большинство пользователей для установки GNU/Linux используют дистрибутивы. Дистрибутив — это не просто набор программ, а ряд решений для разных задач пользователей, объединённых едиными системами установки, управления и обновления пакетов, настройки и поддержки.

Самые распространённые в мире дистрибутивы:

Ubuntu

Быстро завоевавший популярность дистрибутив, ориентированный на лёгкость в освоении и использовании.

openSUSE

Бесплатно распространяемая версия дистрибутива SuSE, принадлежащая компании Novell. Отличается удобством в настройке и обслуживании благодаря использованию утилиты YaST.

Fedora

Поддерживается сообществом и корпорацией RedHat, предшествует выпускам коммерческой версии RHEL.

Debian

Международный дистрибутив, разрабатываемый обширным сообществом разработчиков в некоммерческих целях. Послужил основой для создания множества других дистрибутивов. Отличается строгим подходом к включению несвободного ПО.

Mandriva

Французско-бразильский дистрибутив, объединение бывших Mandrake и Conectiva.

Slackware

Один из старейших дистрибутивов, отличается консервативным подходом в разработке и использовании.

Gentoo

Дистрибутив, собираемый из исходных кодов. Позволяет очень гибко настраивать конечную систему и оптимизировать производительность, поэтому часто называет себя метадистрибутивом. Ориентирован на экспертов и опытных пользователей.

Archlinux

Ориентированный на применение самых последних версий программ и постоянно обновляемый, поддерживающий одинаково как бинарную, так и установку из исходных кодов и построенный на философии простоты «» («Keep it simple, stupid» / «Не усложняй»), этот дистрибутив ориентирован на компетентных пользователей, которые хотят иметь всю силу и модифицируемость Linux, но не в жертву времени обслуживания.

Помимо перечисленных, существует множество других дистрибутивов, как базирующихся на перечисленных, так и созданных с нуля и зачастую предназначенных для выполнения ограниченного количества задач.

Каждый из них имеет свою концепцию, свой набор пакетов, свои достоинства и недостатки. Ни один не может удовлетворить всех пользователей, а потому рядом с лидерами благополучно существуют другие фирмы и объединения программистов, предлагающие свои решения, свои дистрибутивы, свои услуги. Существует множество LiveCD, построенных на основе GNU/Linux, например, Knoppix. LiveCD позволяет запускать GNU/Linux непосредственно с компакт-диска, без установки на жёсткий диск. Большинство крупных дистрибутивов, включая Ubuntu, могут быть использованы как LiveCD.

Для желающих досконально разобраться с GNU/Linux подойдёт любой из дистрибутивов, однако довольно часто для этой цели используются так называемые «source-based» дистрибутивы, то есть предполагающие самостоятельную сборку всех компонентов из исходных кодов, такие как LFS, Gentoo или CRUX.

Область распространения Linux огромна, гораздо больше чем у всех других операционных систем. Кроме того, что Linux прекрасно работает на обычных домашних и рабочих компьютерах и серверах, существуют адаптации Linux к большинству современных процессоров, что позволяет использовать системы с ядром Linux в сетевом оборудовании, домашней «умной» технике, роботах, мобильных телефонах, различных портативных устройствах и другом оборудовании, поддерживающем программируемые операции.

В конечном счёте столь широкий круг поддерживаемых устройств означает превосходную переносимость программ. Например, одно и то же приложение зачастую можно запустить с минимальными усилиями и на обычном компьютере, и на мобильном телефоне на базе Linux. Для примера: Windows и её младший брат Windows Mobile являются полностью несовместимыми платформами.

2 ПЛАТФОРМА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

ЯЗЫК РАЗРАБОТКИ ВИРУСА

Java — строго типизированный объектно-ориентированный язык программирования общего назначения, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle). Разработка ведётся сообществом, организованным через Java Community Process; язык и основные реализующие его технологии распространяются по лицензии GPL. Права на торговую марку принадлежат корпорации Oracle.

Изначально язык назывался Oak («Дуб»), разрабатывался Джеймсом Гослингом для программирования бытовых электронных устройств. Из-за того, что язык с таким названием уже существовал, Oak был переименован в Java[5]. Назван в честь марки кофе Java, которая, в свою очередь, получила наименование одноимённого острова (Ява), поэтому на официальной эмблеме языка изображена чашка с горячим кофе. Существует и другая версия происхождения названия языка, связанная с аллюзией на кофемашину как пример бытового устройства, для программирования которого изначально язык создавался. В соответствии с этимологией в русскоязычной литературе с конца двадцатого и

до первых лет двадцать первого века название языка нередко переводилось как Ява, а не транскрибировалось.

В результате работы проекта мир увидел принципиально новое устройство, карманный персональный компьютер Star7[6], который опередил своё время более чем на 10 лет, но из-за большой стоимости в 50 долларов не смог произвести переворот в мире технологии и был забыт.

Устройство Star7 не пользовалось популярностью, в отличие от языка программирования Java и его окружения. Следующим этапом жизни языка стала разработка интерактивного телевидения. В 1994 году стало очевидным, что интерактивное телевидение было ошибкой.

С середины 1990-х годов язык стал широко использоваться для написания клиентских приложений и серверного программного обеспечения. Тогда же определённое распространение получила технология Java-апплетов графических Java-приложений, встраиваемых в веб-страницы; с развитием возможностей динамических веб-страниц в 2000-е годы технология стала применяться редко.

В веб-разработке применяется Spring Framework; для документирования используется утилита Javadoc.

СРЕДА РАЗРАБОТКИ

IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains.

Первая версия появилась в январе 2001 года и быстро приобрела популярность как первая среда для Java с широким набором интегрированных инструментов для рефакторинга, которые позволяли программистам быстро реорганизовывать исходные тексты программ. Дизайн среды ориентирован на продуктивность работы программистов, позволяя сконцентрироваться на функциональных задачах, в то время как IntelliJ IDEA берёт на себя выполнение рутинных операций.

Начиная с шестой версии продукта IntelliJ IDEA предоставляет интегрированный инструментарий для разработки графического пользовательского интерфейса. Среди прочих возможностей, среда хорошо совместима со многими популярными свободными инструментами разработчиков, такими как CVS, Subversion, Apache Ant, Maven и JUnit. В феврале 2007 года разработчики IntelliJ анонсировали раннюю версию плагина для поддержки программирования на языке Ruby.

Начиная с версии 9.0 среда доступна в двух редакциях: *Community Edition* и *Ultimate Edition*. Community Edition является полностью свободной версией, доступной под лицензией Apache 2.0, в ней реализована полная поддержка Java SE, Kotlin, Groovy, Scala, а также интеграция с наиболее популярными системами управления версиями. В редакции Ultimate Edition, доступной под коммерческой лицензией, реализована поддержка Jakarta EE, UML-диаграмм, подсчёт покрытия кода, а также поддержка других систем управления версиями, языков и фреймворков.

3 ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ РАЗРАБОТКИ ПРОГРАММНОГО ПРОДУКТА

4 ПРОЕКТИРОВАНИЕ ФУНКЦИОНАЛЬНЫХ ВОЗМОЖНОСТЕЙ ПРОГРАММЫ

5 АРХИТЕКТУРА РАЗРАБАТЫВАЕМОЙ ПРОГРАММЫ

ЗАКЛЮЧЕНИЕ

СПИСОК ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ

ПРИЛОЖЕНИЯ